

# PATENT ABSTRACTS OF JAPAN

(11)Publication number : 02-311948

(43)Date of publication of application : 27.12.1990

(51)Int.Cl.

G06F 15/16

G06F 9/38

G06F 11/20

(21)Application number : 01-134301

(71)Applicant : NEC CORP  
NEC ENG LTD

(22)Date of filing : 26.05.1989

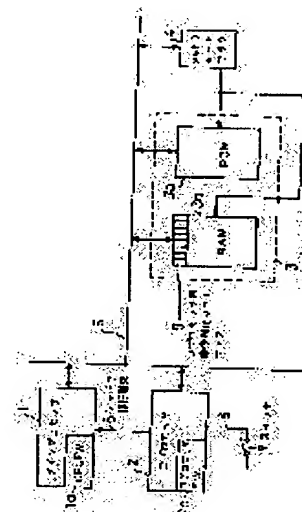
(72)Inventor : ISHII SATOSHI  
NAKAGAWA TAKASHI

## (54) ERROR CORRECTION SYSTEM FOR COPROCESSOR

(57)Abstract:

PURPOSE: To facilitate the bug correction of firmware and the change of the function of a coprocessor by setting an external switch in an error pointing state when the firmware of the coprocessor has an error and turning on the corresponding one of plural soft flags.

CONSTITUTION: A coprocessor 2 decides the state of an external switch 7 via the firmware 2a if the instruction fetched by a main processor 1 is equal to its own one. Then the external switch 7 checks the soft flag corresponding to the instruction via the firmware 2a only when the firmware 2a has an error. When the soft flag is set at the prescribed value, the instruction for its own is not processed and reported to the processor 1. The processor 1 receives the report to change the instruction into a shadow mode and at the same time saves the internal registers. Then the processor 1 carries out the corresponding program of a shadow memory 3 to simulate the instruction of the coprocessor 2 and restores the saved registers. As a result, the error of the firmware 2a can be easily corrected.



## LEGAL STATUS

[Date of request for examination]

[Date of sending the examiner's decision of rejection]

[Kind of final disposal of application other than the examiner's decision of rejection or application converted registration]

[Date of final disposal for application]

[Patent number]

[Date of registration]

[Number of appeal against examiner's decision of rejection]

[Date of requesting appeal against examiner's decision of rejection]

[Date of extinction of right]

Copyright (C); 1998,2003 Japan Patent Office

## ⑫ 公開特許公報(A)

平2-311948

⑬ Int.Cl.<sup>5</sup>G 06 F 15/16  
9/38  
11/20

識別記号

3 7 0 Z  
3 7 0 C  
3 1 0 A

庁内整理番号

6745-5B  
7361-5B  
9072-5B

⑭ 公開 平成2年(1990)12月27日

審査請求 未請求 請求項の数 1 (全6頁)

⑮ 発明の名称 コプロセッサの誤り修正方式

⑯ 特 願 平1-134301

⑰ 出 願 平1(1989)5月26日

⑱ 発 明 者 石 井 智 東京都港区芝5丁目33番1号 日本電気株式会社内  
 ⑲ 発 明 者 中 川 敬 司 東京都港区西新橋3丁目20番4号 日本電気エンジニアリ  
 ング株式会社内  
 ⑳ 出 願 人 日本電気株式会社 東京都港区芝5丁目7番1号  
 ㉑ 出 願 人 日本電気エンジニアリ 東京都港区西新橋3丁目20番4号  
 ング株式会社  
 ㉒ 代 理 人 弁理士 境 廣 巳

## 明 細 書

## 1. 発明の名称

コプロセッサの誤り修正方式

## 2. 特許請求の範囲

各々ファームウェアで制御されるメインプロセッサとそのコプロセッサとを有し且つシャドウメモリ方式を採用した情報処理装置において、

前記コプロセッサのファームウェアにおける誤りの存在する部分によって処理されるコプロセッサ用命令の機能をシミュレートするプログラムを記憶させたシャドウメモリと、

前記コプロセッサのファームウェアの誤りの有無を示す外付スイッチと、

各コプロセッサ用命令に対応したソフトフラグとを設け、

前記コプロセッサは、前記メインプロセッサでフェッチされた命令が自己用の命令のとき先ず前記外付スイッチの状態をファームウェアにて判定し、前記外付スイッチがファームウェアに誤りが有ることを示す場合に限り、前記命令に対応する

ソフトフラグをファームウェアにてチェックし、所定の値であれば前記自己用の命令を処理することなく前記メインプロセッサに通知し、

前記メインプロセッサは前記通知を受けることによりシャドウモードに変更すると共に内部レジスタ類の退避を行った後、前記シャドウメモリの該当するプログラムを実行することにより前記コプロセッサの命令をシミュレートし、その後前記退避した内部レジスタ類の復元を行うと共にノーマルモードに変更して処理を続行することを特徴とするコプロセッサの誤り修正方式。

## 3. 発明の詳細な説明

(産業上の利用分野)

本発明は、各々ファームウェアで制御されるメインプロセッサとそのコプロセッサとを有する情報処理装置においてコプロセッサのファームウェアの機能の一部に存在する誤りを修正する方式に関するものである。

(従来の技術)

プロセッサのファームウェアがLSIで構成さ

れた実行ユニット内部に格納されている如くファームウェア自体の変更が容易でない状況において、そのファームウェアの機能の一部に存在する誤りを修正する方式としては、例えば特開昭63-98034号公報に見られるようなシャドウメモリ方式を採用した情報処理装置が知られている。

第3図はそのような従来の情報処理装置の要部ブロック図であり、ファームウェア制御によるメインプロセッサ11とそのバス(内部バス)15に接続されたシャドウメモリ13及びシャドウモードフラグ14とを含んでいる。なお、シャドウメモリ13はソフトフラグ12を有するRAM13aとROM13bとを含み、メインプロセッサ11にはそのファームウェア制御にかかるファームウェア(CPUFW)11aが設けられている。

一般にシャドウメモリ方式とは、メモリの或る部分をフラグによりメモリ空間とシャドウメモリ空間に分け、システム立ち上げやシステム固有のコントロール時に、シャドウモードに切り替えてシャドウメモリに格納されているプログラム(シ

ャドウプログラム)を実行し、その終了時、シャドウモードからノーマルモードに切り替え、通常はシャドウメモリがアクセスできないようになっているものである。第3図では、そのような既存のシャドウメモリ方式を改良し、メインプロセッサ11中のファームウェア11aの誤りを修正する機能を既存のシャドウメモリ方式中に導入したものである。即ち、メインプロセッサ11におけるファームウェア11a中の誤りの存在する部分によって処理される命令や処理ルーチンをシミュレートするプログラムをシャドウメモリ13に記憶させると共に、メインプロセッサ11の各命令に対応し且つ誤りの存在する換言すればシミュレートしたい命令に対応するフラグは事前にONにしたソフトフラグ12を設け、メインプロセッサ11は、命令をフェッチする毎にその命令に対応するソフトフラグ12のフラグ値をファームウェア11aにて判定し、シミュレートすべきことを示すONである場合にはファームウェア11aによってその命令を処理することなくシャドウモ

ドに変更し、内部レジスタ類の退避を行った後、シャドウメモリ13の該当するプログラムを実行して前記命令をシミュレートし、その後前記退避した内部レジスタ類の復元を行うと共にノーマルモードに変更して次の命令をフェッチする処理に戻るように構成されているものである。前記シミュレート用のプログラムは通常のソフトウェア命令以外にプロセッサの内部資源を制御できる特別命令をも使用して作成されており、通常のソフトウェア命令のみで構成されたシミュレート用のプログラムに比べて、そのシミュレートを容易にし且つ実行速度を上げている。

(発明が解決しようとする課題)

第3図に示した従来の方式によって、メインプロセッサ11のファームウェア11aに存在する誤りを実質的に修正することが可能である。

ところで、情報処理装置の中にはメインプロセッサ以外にそのコプロセッサを有し、高度な演算等を高速に実行し得るようにしたものがある。このようなコプロセッサを有する情報処理装置にお

いて、若しコプロセッサのファームウェアの一部に誤りがあると、それによって処理されるコプロセッサ用命令や各種処理ルーチンの実行が不可能となる。従って、その誤りを修正する必要があるが、コプロセッサのファームウェアがLSI内部に格納されている場合には、やはりその変更は容易ではない。また、コプロセッサはメインプロセッサと異なり特殊なプロセッサなので、第3図の従来の方式をそのままコプロセッサに適用することは困難である。

本発明はこのような事情に鑑みて為されたものであり、各々ファームウェアで制御されるメインプロセッサとそのコプロセッサとを有し且つシャドウメモリ方式を採用した情報処理装置において、コプロセッサのファームウェアの誤りをそのファームウェア自体を変更することなく修正することのできるコプロセッサの誤り修正方式を提供することにある。

(課題を解決するための手段)

本発明は上記の目的を達成するために、各々フ

ファームウェアで制御されるメインプロセッサとそのコプロセッサとを有し且つシャドウメモリ方式を採用した情報処理装置において、前記コプロセッサのファームウェアにおける誤りの存在する部分によって処理されるコプロセッサ用命令の機能をシミュレートするプログラムを記憶させたシャドウメモリと、前記コプロセッサのファームウェアの誤りの有無を示す外付スイッチと、各コプロセッサ用命令に対応したソフトフラグとを設け、前記コプロセッサは、前記メインプロセッサでフェッチされた命令が自己用の命令のとき先ず前記外付スイッチの状態をファームウェアにて判定し、前記外付スイッチがファームウェアに誤りがあることを示す場合に限り、前記命令に対応するソフトフラグをファームウェアにてチェックし、所定の値であれば前記自己用の命令を処理することなく前記メインプロセッサに通知し、前記メインプロセッサは前記通知を受けることによりシャドウモードに変更すると共に内部レジスタ類の退避を行った後、前記シャドウメモリの該当するプログ

ラムを実行することにより前記コプロセッサの命令をシミュレートし、その後前記退避した内部レジスタ類の復元を行うと共にノーマルモードに変更して処理を続行するように構成されている。

#### (作用)

本発明のコプロセッサの誤り修正方式においては、シャドウメモリが、前記コプロセッサのファームウェアにおける誤りの存在する部分によって処理されるコプロセッサ用命令の機能をシミュレートするプログラムを保持し、且つ、外付スイッチが前記コプロセッサのファームウェアの誤りの有無を示すと共に、各コプロセッサ用命令に対応したソフトフラグがその対応するコプロセッサ用命令がシミュレートの対象になっているか否かを示し、メインプロセッサでフェッチされた命令がコプロセッサ用命令のとき、コプロセッサが、先ず前記外付スイッチの状態をファームウェアにて判定し、そのスイッチがファームウェアに誤りがあることを示す場合に限り、その命令に対応するソフトフラグをファームウェアにてチェックし、

所定の値であればその命令を処理することなく前記メインプロセッサに通知し、この通知を受けたメインプロセッサが、シャドウモードに変更すると共に内部レジスタ類の退避を行った後、前記シャドウメモリの該当するプログラムを実行することにより前記コプロセッサの命令をシミュレートし、その後前記退避した内部レジスタ類の復元を行うと共にノーマルモードに変更して処理を続行する。

#### (実施例)

次に、本発明の実施例について図面を参照して詳細に説明する。

第1図は本発明のコプロセッサの誤り修正方式の一実施例の処理の流れ図、第2図は本発明を適用した情報処理装置の一例を示す要部ブロック図である。本発明を適用した情報処理装置の一例は第2図に示すように、内部のファームウェア(CPUFW)1aで制御されるメインプロセッサ1と、同じく内部のファームウェア(コプロセッサFW)2aで制御されるコプロセッサ2と、RAM

M3aおよびROM3bを有するシャドウメモリ3と、シャドウモードフラグ4と、これらを接続するバス(内部バス)5とを含んでいる。

コプロセッサ2には、信号線8を介してスイッチ(外付スイッチ)7が接続されている。このスイッチ7はコプロセッサ2のファームウェア(コプロセッサFW)2aに誤りが全くない場合はOFFにされ、1箇所でも誤りが存在する場合にはONにされるスイッチであり、コプロセッサ2のファームウェア2aはファームウェアの1ステップでスイッチ7の状態を判定できるようになっている。

シャドウメモリ3のRAM3a内には、各コプロセッサ用命令に1対1で対応する複数のフラグから成るソフトフラグ(コプロセッサ用命令対応ソフトフラグ)9が設けられており、コプロセッサ2のファームウェア2aに1つも誤りが存在しない場合にはソフトフラグ9の全てのフラグはOFFにされている。しかし、或るファームウェア部分に誤りがあって、それが原因で或るコプロセ

ッサ用命令やコプロセッサのファームウェアの或る処理ルーチンが正しく実行されない場合には、そのコプロセッサ用命令に対応するフラグがONにされる。そして、その正しく実行されないコプロセッサ用命令をメインプロセッサ1がシミュレートするために必要となるシミュレート用のプログラムが、シャドウメモリ3に事前に登録されている。

次に、第1図および第2図を参照して、本実施例の動作を説明する。なお、第1図において破線で囲んだ処理はコプロセッサ2内の処理を示し、他の処理はメインプロセッサ1の処理を示す。

ノーマルモードにおいて、メインプロセッサ1は1つの命令をフェッチする毎に(S1)、その命令がコプロセッサ用命令であるか否かを判定する(S2)。コプロセッサ用命令でなければ、メインプロセッサ1自身がその命令を実行し(S3)、命令フェッチ処理S1に戻って次の命令を処理する。

処理S2でコプロセッサ用命令と判定された場

セットし(S15)、これをメインプロセッサ1に伝達する。しかし、OFFの場合には、そのコプロセッサ用命令はシミュレートの対象になっているので、ファームウェア2aによってその命令を実行せず、シャドウモード要求コマンドをセットし(S16)、これをメインプロセッサ1に伝達する。即ち、第2図に示すようにシャドウ移行要求6をメインプロセッサ1に出す。

メインプロセッサ1は、コプロセッサ2からのコマンドを解析してシャドウモード要求コマンドか否かを判定し(S4)、シャドウモード要求コマンドでなければ制御を処理S1に戻し、次の命令をフェッチする。しかし、シャドウモード要求コマンドの場合には、現在のモードをシャドウモードに変更する為の処理を行い(S5)、自メインプロセッサ1内のレジスタ類をシャドウメモリ3のRAM3aにセーブする(S6)。そして、以下のようにしてコプロセッサ用命令にかかるシミュレートを実行する(S7~S9)。

先ず、必要に応じてコプロセッサ2からメイン

合、そのコプロセッサ用命令はコプロセッサ2に伝達される。コプロセッサ2では、先ずスイッチ7の状態がONかOFFかをそのファームウェア2aにて判定し(S12)、OFFであれば処理S14へ、ONであれば処理S13へ進む。

処理S14へ進んだ場合すなわちコプロセッサ2のファームウェア2aに1箇所も誤りが存在しない場合、コプロセッサ2のファームウェア2aの処理によってその命令を実行し、次に通常終了コマンドをセットし(S15)、これをメインプロセッサ1に伝達する。

他方、処理S13に進んだ場合すなわちコプロセッサ2のファームウェア2aに少なくとも1箇所誤りがある場合、コプロセッサ2は、コプロセッサ用命令に対応するソフトフラグ9のフラグ値がONかOFFかをチェックする(S13)。OFFの場合、そのコプロセッサ用命令についてはシミュレートの対象になっていないので、コプロセッサ2のファームウェア2aの処理によってその命令を実行し(S14)、通常終了コマンドを

プロセッサ3へデータを転送する(S7)。ここで、データとはシミュレートしようとする命令が使用するデータのうちコプロセッサ2のレジスタ等に保持されているデータをいう。次にメインプロセッサ1はシャドウメモリ3に記憶された該当するシャドウプログラムを実行する(S8)。次に、このシャドウプログラムの実行により得られたデータ即ち本来コプロセッサ2がその命令を実行することにより得るデータを、メインプロセッサ1からコプロセッサ2へ転送する(S9)。これで、コプロセッサ2の状態はそのコプロセッサ用命令を恰も自らのファームウェア2aが実行したと同じ状態になる。

メインプロセッサ1は上記のシミュレートを終わると、シャドウメモリ3に退避していた自己のレジスタ類をリストアし(S10)、シャドウモードをリセットしてノーマルモードに変更し(S11)、次の命令のフェッチに移行する(S1)。【発明の効果】

以上説明したように、本発明のコプロセッサの

誤り修正方式においては、コプロセッサのファームウェアの誤りの有無を示す外付スイッチと、各コプロセッサ用命令に対応したソフトフラグを設け、コプロセッサのファームウェアに誤りが存在する場合には外付スイッチをそれを指示する状態に設定しておくと共に、ソフトフラグ中の該当するフラグをONにしておくことにより、そのフラグに対応するコプロセッサ用命令をコプロセッサ中のファームウェアで処理せずにシャドウメモリに予め記憶させたプログラムを使用してメインプロセッサでシミュレートすることができ、コプロセッサにおけるファームウェアのバグ修正や機能変更をそのファームウェア自体を変更することなく容易に行うことができる効果がある。

また、外付スイッチを設けず、常にソフトフラグをチェックする構成にすれば、コプロセッサのファームウェアの誤りが後に修正されて誤りが全くなくなった場合におけるオーバーヘッドが大きくなるが、本発明のように外付スイッチを設ければ、誤りが全て無くなったときソフトフラグを一

切チェックする必要がなくなり、性能低下を極端かにすることができる。よって、コプロセッサのファームウェアのバグ等の修正後、第1図の処理S12、S13等のチェックルーチンを削除する必要がなくコプロセッサのファームウェアの評価期間を短縮することができる効果もある。

#### 4. 図面の簡単な説明

第1図は本発明の一実施例の処理の流れ図、

第2図は本発明を適用した情報処理装置の一例を示す要部ブロック図および、

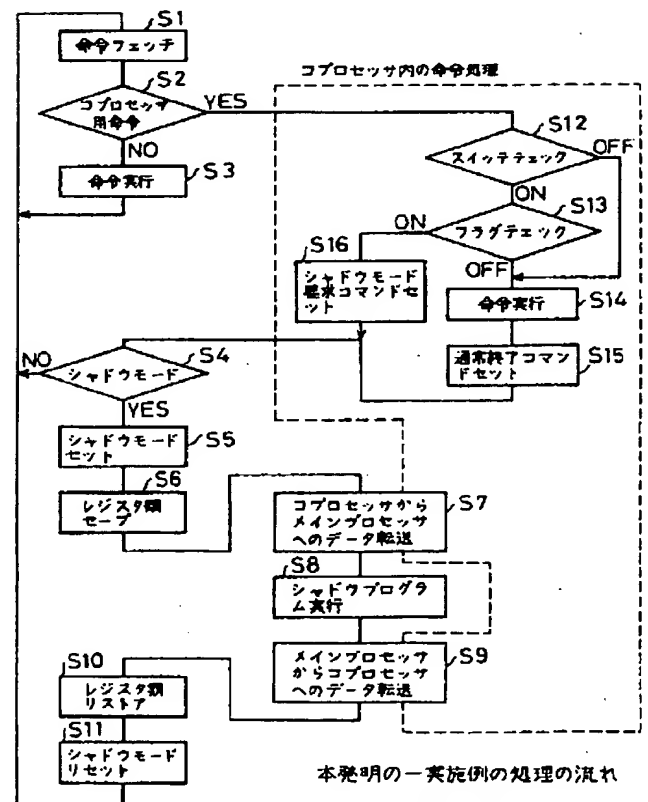
第3図は従来方式のブロック図である。

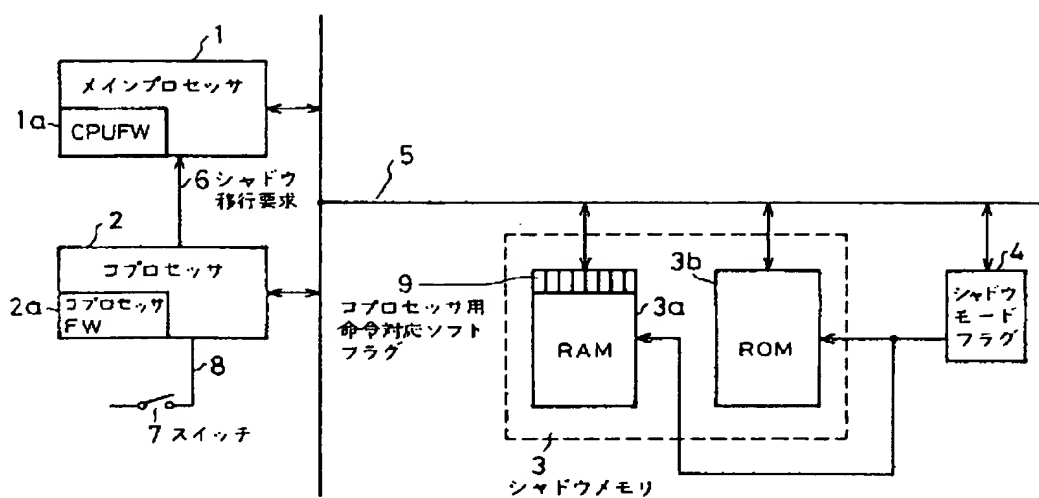
図において、

- 1…メインプロセッサ
- 1a…メインプロセッサのファームウェア
- 2…コプロセッサ
- 2a…コプロセッサのファームウェア
- 3…シャドウメモリ
- 3a…RAM
- 3b…ROM
- 4…シャドウモードフラグ

- 5…バス（内部バス）
- 6…シャドウ移行要求
- 7…スイッチ（外付スイッチ）
- 8…信号線
- 9…ソフトフラグ

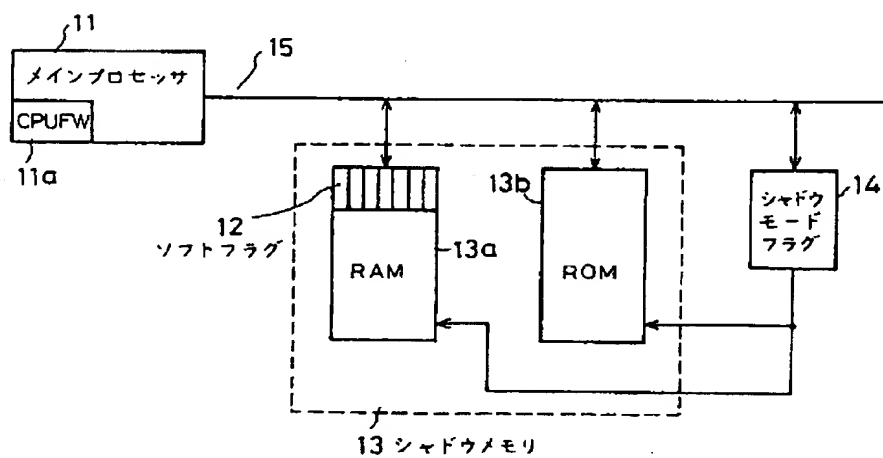
特許出願人 日本電気株式会社外1名  
代理人 弁理士 境 廣 巳





本発明を適用した情報処理装置の一例

第 2 図



従来方式のブロック図

第 3 図